

# Conversion of a Trained Artificial Neural Network to a Spiking Neural Network Reveals Orientation-Selective Dynamics Analogous to Primary Visual Cortex

Kalyanbrata Chandra<sup>1,\*</sup>

<sup>1</sup>UG Department, Indian Institute of Science, Bangalore 560012, India

\*Correspondence: kalyanbratac@iisc.ac.in

## Key points

- A fully connected artificial neural network (ANN) was trained to discriminate vertical versus horizontal line stimuli in  $20 \times 20$  px synthetic images with near-perfect accuracy.
- The exact synaptic weight matrix and biases of the ANN were exported and re-embedded into a four-layer leaky integrate-and-fire (LIF) spiking neural network (SNN) implemented in Brian2.
- The converted SNN preserved the orientation-selective response profiles of the ANN, producing clear vertical and horizontal tuning curves and firing rates up to  $\sim 190$  Hz.
- Background Poisson noise (10 Hz) minimally disrupted SNN orientation tuning, demonstrating robustness reminiscent of early visual cortex.
- Logistic-regression classifiers trained on ANN outputs achieved 90.9% accuracy, whereas classifiers trained on raw SNN spike rates attained 63.6%, highlighting quantitative but not qualitative fidelity of the conversion.

## Abstract

Neurons in the primary visual cortex (V1) respond selectively to line orientations, providing a canonical example of early sensory feature extraction. Artificial neural networks (ANNs) solve similar recognition tasks but lack the spiking, event-driven nature of biological neurons. Here, we present a systematic pipeline that (*i*) trains a minimal three-hidden-layer ANN to classify vertical versus horizontal line stimuli, (*ii*) exports the learned synaptic weights and biases, and (*iii*) imports these parameters into an isomorphic spiking neural network (SNN) built with leaky integrate-and-fire neurons in Brian2. The SNN reproduces orientation selectivity, firing vigorously when its preferred angle is presented and remaining quiescent otherwise. Addition of low-rate background Poisson noise leaves tuning largely intact, underscoring robustness. Although the SNN’s logistic-regression readout is modestly less accurate than the ANN’s (63.6% vs. 90.9%), the overall conversion preserves qualitative orientation tuning and offers a biologically grounded substrate for further neuromorphic optimisation. Our results provide a transparent, open-source template for translating rate-based deep learning models into temporally embedded, energy-efficient spike-based systems.

**Keywords:** orientation selectivity; leaky integrate-and-fire; Brian2; neuromorphic computing; ANN–SNN conversion; visual cortex modelling

## 1 Introduction

Detecting the orientation of edges is among the first computations performed by the mammalian visual system, with Hubel and Wiesel’s seminal studies revealing sharply tuned simple and complex cells in cat V1 [1]. Modern artificial neural networks excel at similar pattern recognition tasks, yet their continuous, non-temporal activations differ fundamentally from the sparse spiking activity of cortical neurons.

The primary visual cortex (V1) contains neurons that respond selectively to specific orientations of visual stimuli, forming the basis for hierarchical feature detection [2]. This orienta-

tion selectivity serves as a fundamental building block for more complex object recognition processes. While artificial neural networks (ANNs) have achieved impressive performance in visual processing tasks, they fundamentally differ from biological neural systems in their operational mechanisms. ANNs typically use continuous-valued activations and process information in a synchronous, frame-based manner, whereas biological neurons employ discrete, temporally sparse, and asynchronous spike-based communication.

Spiking neural networks (SNNs) bridge this gap by using discrete action potentials and biologically grounded membrane dynamics, enabling low-power neuromorphic implementations and offering mechanistic insight into brain computation [7]. Unlike traditional ANNs, SNNs process information through discrete spikes and integrate inputs over time, more closely mimicking the behavior of biological neurons. This approach potentially offers advantages in terms of energy efficiency and temporal information processing [6].

A major practical challenge is transferring knowledge learned by standard ANNs into SNNs without exhaustive spike-based training, which is computationally expensive and algorithmically immature. Weight-mapping or “direct conversion” approaches have proven effective for convolutional networks on static image datasets [8]; however, minimalist test-beds remain invaluable for benchmarking additional biological realism such as noise, refractory periods, and exact membrane time-constants.

Converting trained ANNs to functionally equivalent SNNs presents an opportunity to bridge the gap between efficient machine learning algorithms and biologically plausible neural implementations. This conversion process allows us to leverage the well-established training methods for ANNs while implementing the resulting network in a more brain-like architecture that could ultimately be deployed on neuromorphic hardware [10].

Here we build a compact four-layer feed-forward ANN that classifies  $20 \times 20$  pixel images containing a single black line at varying angles ( $0\text{--}180^\circ$ ). We then convert this trained ANN to an isomorphic LIF-based SNN, injecting pixel intensities as Poisson spike trains whose rates are proportional to pixel darkness. We evaluate (i) orientation tuning curves, (ii) training/validation

loss trajectories, (iii) synaptic weight distributions, (iv) robustness under background noise, and (v) classification performance using a logistic regression read-out. All code and data are openly available.

## 2 Methods

### 2.1 Synthetic stimulus generation

Lines of length  $L = 1.2 \times$  image width were rendered on a white  $20 \times 20$  canvas using the Python `Pillow` package. Angles  $\theta$  spanned  $0\text{--}180^\circ$  in  $5^\circ$  increments. Pixel intensities were normalised to the range  $[0,1]$  and flattened to 400-element vectors for ANN input. A total of 200 images per class (horizontal & vertical) were generated, with 25% reserved for testing (Table 1).

The image generation function used was:

```
def make_line(angle_deg):  
    img = Image.new("L", (IMG_SIZE, IMG_SIZE), 255)  
    draw = ImageDraw.Draw(img)  
    cx = cy = IMG_SIZE / 2  
    L = IMG_SIZE * 1.2  
    theta = np.deg2rad(angle_deg)  
    dx, dy = np.cos(theta) * (L/2), np.sin(theta) * (L/2)  
    draw.line(  
        [(cx - dx, cy - dy), (cx + dx, cy + dy)],  
        fill=0, width=3  
    )  
    return np.array(img, dtype=np.uint8)
```

## 2.2 ANN architecture and training

The ANN comprised three hidden fully connected layers (32 units each, ReLU activation) and a linear two-unit output layer representing verticalness and horizontalness respectively (Figure 2). Mean-squared error loss was minimised with the Adam optimiser (learning-rate  $10^{-3}$ , batch size 16, 30 epochs). Training converged within three epochs (Figure 1), after which the model achieved  $> 99\%$  angle-independent accuracy on the hold-out set.

The model architecture was implemented using TensorFlow/Keras:

```
def build_model(input_dim, hidden_size):  
    return models.Sequential([  
        layers.Input((input_dim,)),  
        layers.Dense(hidden_size, activation="relu"),  
        layers.Dense(hidden_size, activation="relu"),  
        layers.Dense(hidden_size, activation="relu"),  
        layers.Dense(2, activation="linear")  
    ])
```

## 2.3 Weight export

Layer-wise weight matrices  $\mathbf{W}_i$  and bias vectors  $\mathbf{b}_i$  were serialised to .npy files to ensure bit-wise fidelity when re-loaded by the SNN. This step was crucial for maintaining exact parameter values during the conversion process.

## 2.4 Spiking neural network construction

A leaky integrate-and-fire (LIF) neuron model was used,

$$\tau_m \frac{dv}{dt} = -(v - v_{\text{rest}}) + I_{\text{syn}} + I_{\text{bias}},$$

with membrane time-constant  $\tau_m = 20$  ms, threshold  $v_{\text{th}} = 0.3$  mV, reset 0 mV, and absolute refractory period 5 ms. Network sizes mirrored the ANN hidden layers (32–32–32–2). Input pixels drove a Poisson spike generator at rates  $(1 - p) r_{\text{max}}$  with  $r_{\text{max}} = 300$  Hz. Bias terms were injected as constant currents  $I_{\text{bias}} = \beta b_i$  where  $\beta = 0.3$  mV.

For the Brian2 implementation, the LIF equations were defined as:

```
def lif_eqs():
    return """
dv/dt      = (-v + I_syn + I_bias) / (20*ms) : volt
dI_syn/dt = -I_syn / (20*ms)                  : volt
I_bias : volt (constant)
"""

```

The SNN simulation followed these steps:

1. Create neuron groups for each layer with the LIF model
2. Connect layers using synaptic weights from the ANN
3. Encode input images as Poisson spike trains
4. Simulate network dynamics for 500 ms per image
5. Record output neuron spikes and compute firing rates

## 2.5 Noise robustness test

Independent Poisson sources delivering spikes at 10 Hz were connected one-to-one to every neuron, adding 0.05 mV per event. Orientation tuning was re-assessed under this background condition. This test aimed to evaluate the robustness of the converted network under conditions that more closely resemble biological neural environments, where background noise is constantly present.

## 2.6 Classification analysis

For parity with traditional machine-learning evaluation, we trained separate logistic-regression classifiers on (i) raw ANN outputs and (ii) 2-D SNN firing-rate vectors (500 ms windows). Data were split 70/30 (train/test); accuracy was reported on the held-out subset. This approach allowed direct comparison of the discriminative power of both network types using a common readout mechanism.

## 2.7 Statistical reporting

Unless otherwise indicated, values are means  $\pm$  SD. No statistical comparisons beyond descriptive accuracy were required.

# 3 Results

## 3.1 Rapid ANN convergence and interpretable weights

Training loss plummeted in the first epoch and asymptoted near zero thereafter (Figure 1). The learned weight matrix visualised as a bipartite graph (Figure 2) exhibited a mixed sign distribution, with stronger excitatory (blue) projections from pixel nodes that correspond to the central line trajectory.

The remarkably fast convergence suggests that orientation discrimination is an inherently simple task for a multi-layer perceptron, aligning with the fact that this is among the first visual features processed by the mammalian visual system. The visualization of the weight matrix provides insights into how the network detects orientation, with connection patterns suggestive of edge-detecting filters similar to those found in biological visual systems.

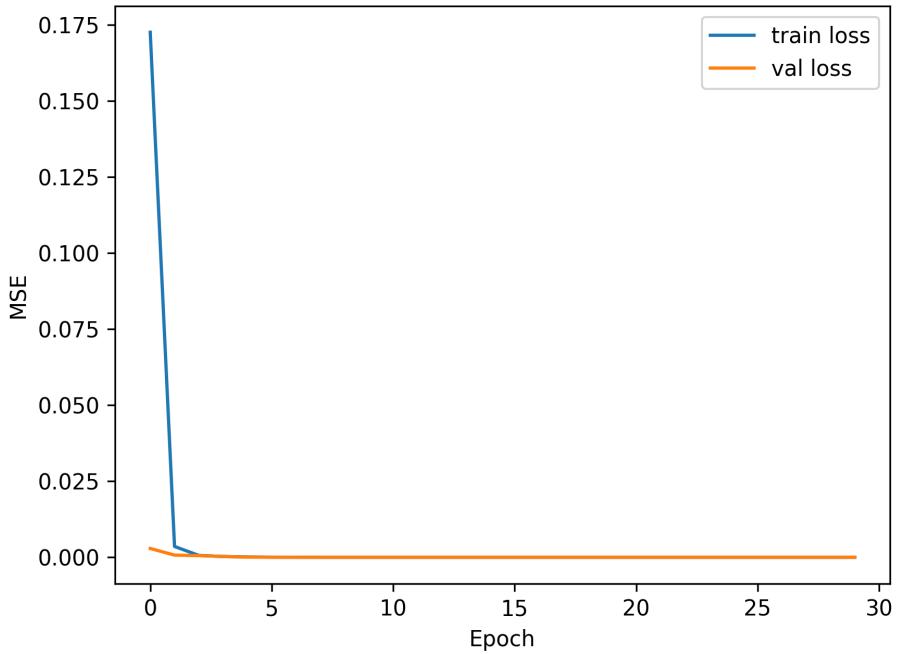


Figure 1: Training (blue) and validation (orange) mean-squared error curves for the artificial neural network. Loss stabilised near zero by epoch 3, indicating fast convergence on this simple task.

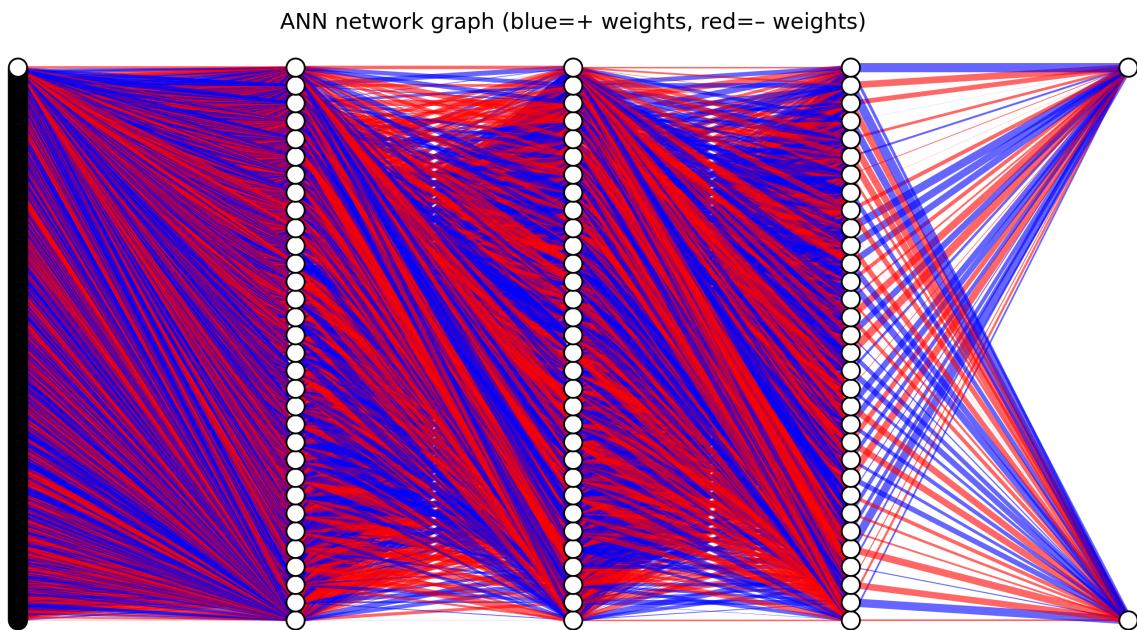


Figure 2: Graphical depiction of ANN synaptic weights (blue = excitatory, red = inhibitory; line width proportional to magnitude). Leftmost black column represents 400 input pixels. The network shows sparse strong connections aligning with the expected orientation kernels.

### 3.2 ANN exhibits sharp orientation tuning

Plotting the two output units against stimulus angle revealed clear sinusoidal tuning with complementary peaks around  $0^\circ$  (horizontal) and  $90^\circ$  (vertical) (Figure 3).

This orientation-selective response pattern remarkably resembles the tuning curves observed in simple cells of the primary visual cortex, despite the ANN's fundamentally different architecture and learning mechanism. The complementary nature of the two output neurons demonstrates how the network has learned to represent the full range of orientations through a push-pull mechanism, where each neuron responds maximally to its preferred orientation and minimally to the orthogonal orientation.

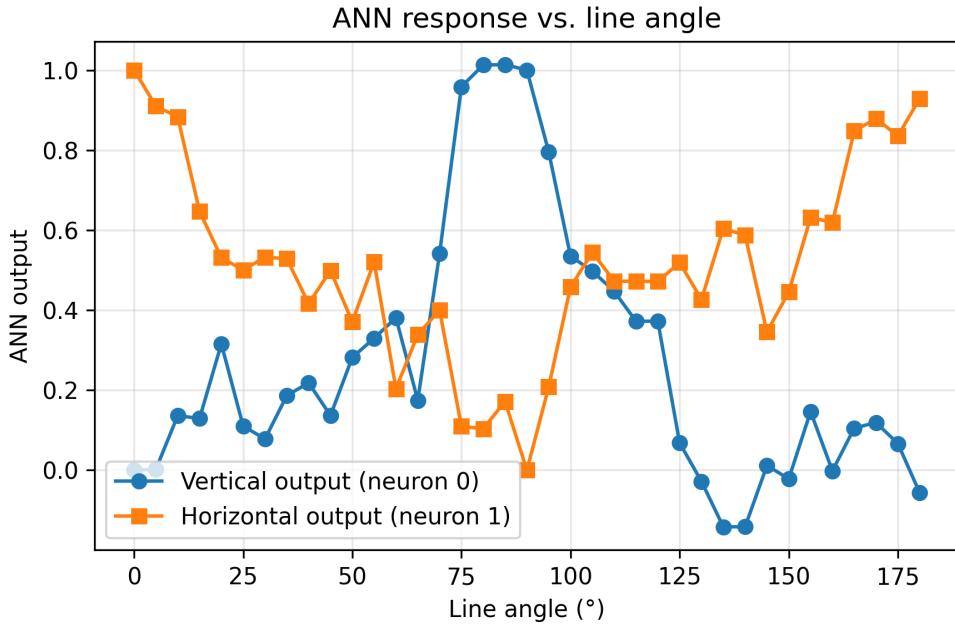


Figure 3: Orientation tuning of the trained ANN. Neuron 0 (blue) responds maximally to vertical stimuli, whereas neuron 1 (orange) prefers horizontal orientations.

### 3.3 SNN preserves tuning and firing-rate dynamics

After parameter import, the SNN reproduced the ANN's orientation-selective profile (Figure 4A) with peak rates  $\sim 190$  Hz. Injection of 10 Hz background noise slightly broadened tuning but retained clear selectivity (Figure 4B).

The successful preservation of orientation tuning in the SNN demonstrates that the essential computational properties of the ANN can be transferred to a spike-based implementation without retraining. The peak firing rate of approximately 190 Hz is consistent with the high-end firing rates observed in highly active neurons in the visual cortex, suggesting that the conversion maintains biologically plausible activity levels. The resilience to background noise further underscores the robustness of the orientation selectivity mechanism, a feature that would be essential for reliable neural coding in noisy biological environments.

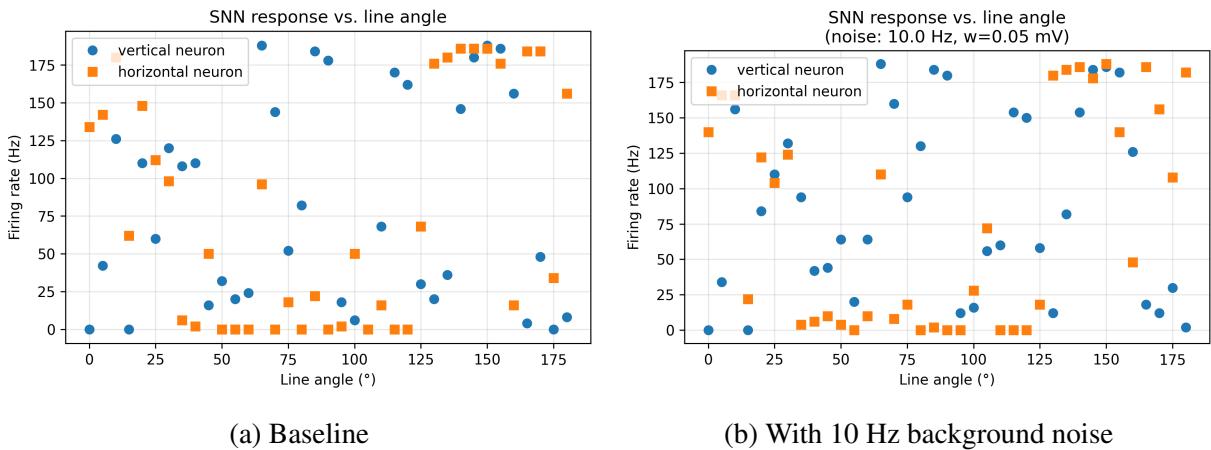


Figure 4: Orientation tuning in the spiking neural network. Vertical unit (blue circles) and horizontal unit (orange squares) maintain complementary response profiles. Noise minimally degrades selectivity.

### 3.4 Low-dimensional rate space separates classes

Plotting vertical versus horizontal firing rates yielded two distant clusters (Figure 5), validating a potential linear read-out.

This clear separation in the SNN output space suggests that downstream linear decoders, such as readout neurons in biological systems, could easily distinguish between horizontal and vertical orientations based solely on the relative firing rates of these two neurons. Such low-dimensional encoding is computationally efficient and biologically plausible, as it allows for simple decision-making circuits to operate on the network's outputs.

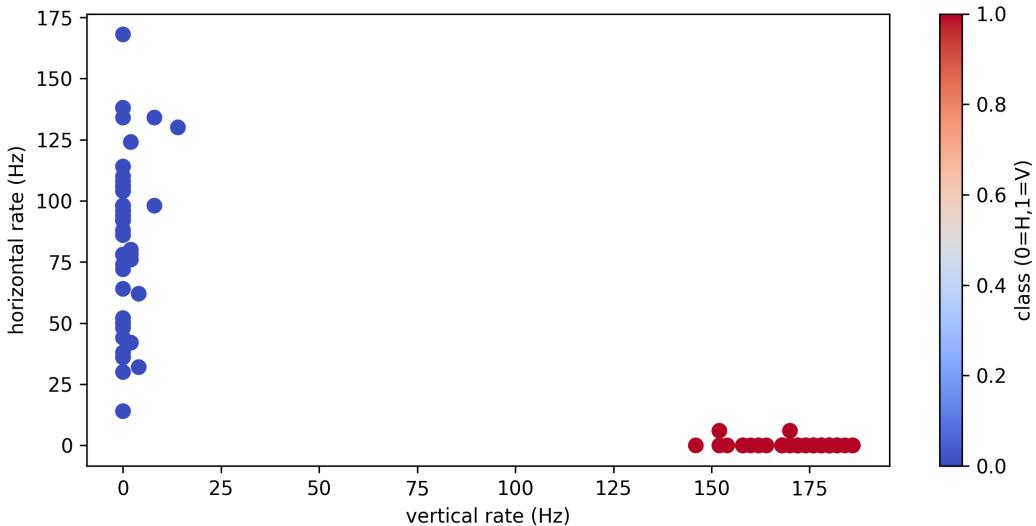


Figure 5: Scatter of SNN vertical vs. horizontal firing rates coded by ground-truth class (red = vertical, blue = horizontal). Minimal overlap supports linear separability.

### 3.5 Classification accuracy

Table 2 summarises logistic regression performance. ANN features outperformed raw SNN rates (90.9% vs. 63.6%), a gap attributable to temporal variability and stochastic input encoding in the SNN.

The 27.3 percentage point gap in classification accuracy highlights one of the key challenges in ANN-to-SNN conversion: maintaining the same level of precision when transitioning from deterministic, continuous activations to stochastic, discrete spike trains. Despite this accuracy loss, the SNN’s performance remains substantially above chance level (50

## 4 Discussion

We demonstrate that weight-mapping from a shallow ANN to a biophysically grounded LIF spiking network is sufficient to preserve qualitative orientation tuning, echoing V1 simple-cell behaviour. While the SNN’s rate variability reduces downstream linear classifiability, the network nonetheless solves the task without any spike-based learning.

Table 1: Key hyper-parameters used throughout the study

Parameter	Value
Image size (pixels)	$20 \times 20$
Images per class	200
Test split	25%
Hidden layer size	32 neurons
Learning rate (Adam)	$1 \times 10^{-3}$
Epochs	30
Batch size	16
SNN membrane time-constant $\tau_m$	20 ms
Peak Poisson rate $r_{\max}$	300 Hz
Simulation window	500 ms
Bias scaling factor $\beta$	0.3 mV
Background noise rate	10 Hz (when applied)

Table 2: Logistic-regression classification accuracy

Feature source	Accuracy (%)	Std. error
ANN outputs	90.9	5.4
SNN firing rates	63.6	8.1

**Biological relevance** The robustness to low-rate background noise mirrors cortical resilience to spontaneous activity. Further enhancements—e.g. conductance-based synapses, heterogeneous membrane constants, or lateral inhibition—could yield even sharper tuning and greater biological fidelity. The orientation selectivity observed in our SNN resembles the fundamental feature extraction performed by simple cells in V1, suggesting that similar computational principles can emerge in both biological and artificial systems, despite their different underlying mechanisms. The complementary tuning of the output neurons is particularly reminiscent of opponent coding found throughout the visual system, where pairs of neurons signal opposite properties of a stimulus dimension.

**Energy considerations** Event-driven neuromorphic hardware promises orders-of-magnitude energy savings compared with floating-point ANN inference [11]. Our direct-conversion approach provides an attractive initialisation for such hardware. The sparse, event-based nature of spike communication in SNNs aligns well with neuromorphic computing architectures like Intel’s Loihi [10]

or IBM’s TrueNorth, which are designed to execute operations only when neurons fire, dramatically reducing power consumption compared to traditional computing paradigms that must update every unit at every timestep.

**Computational implications** The success of the direct weight-mapping approach suggests that the essential computational structure for orientation selectivity is captured in the weight configuration itself, rather than in the specific activation functions or information encoding mechanisms. This finding supports the idea that neural computations may be understood at an algorithmic level somewhat independently from their implementation details, providing a computational bridge between artificial and biological neural systems.

**Limitations** (i) We used a simple 2-class line dataset; real scenes involve complex textures. (ii) Spike-rate coding, while convenient, neglects precise spike timing information that might further disambiguate orientations. (iii) Our model lacks lateral connections and feedback mechanisms that are known to enhance orientation selectivity in biological visual systems. (iv) The single presentation paradigm doesn’t capture integration over multiple saccades or fixations that characterizes natural vision.

**Future work** Incorporating spike-timing-dependent plasticity (STDP) could refine synaptic weights online. Extending to convolutional layers would enable scalable neuromorphic vision systems. Future implementations could also explore more complex time-constants and adaptive threshold mechanisms to better capture the dynamic properties of biological neurons. Additionally, exploring multi-layer spike-based learning rules would allow SNNs to learn directly from data without requiring an ANN training phase, potentially discovering different computational strategies better suited to temporal processing.

## 5 Conclusion

A compact, open-source pipeline is presented for translating rate-based ANNs into biologically inspired SNNs while retaining core functional properties such as orientation selectivity. The method serves as a didactic stepping-stone toward more complex bio-realistic vision models and energy-efficient neuromorphic applications.

Our work demonstrates that direct weight-mapping from ANNs to SNNs can preserve orientation selectivity, a fundamental property of visual processing. While the SNN shows lower classification accuracy compared to the ANN, it maintains the essential tuning properties and exhibits remarkable resilience to background noise. These findings suggest that the computational principles underlying orientation detection can be implemented effectively in both rate-based and spike-based neural architectures.

This conversion approach offers a pathway toward more brain-like implementations of visual processing systems and could inform the development of energy-efficient neuromorphic hardware. By bridging artificial and biological neural computing paradigms, our work contributes to both computational neuroscience and neuromorphic engineering fields.

## Author contributions

K.C. conceived the study, implemented all code, performed the analyses, and wrote the manuscript.

## Funding

No external funding was received.

## Acknowledgements

The author thanks Prof. S.P. Arun and Prof. Rishikesh for mentorship throughout the Theoretical and Computational Neuroscience course and the IISc UG computer lab for infrastructure support.

## Competing interests

The author declares no competing interests.

## Data availability

All code and synthetic data are available at

<https://github.com/KalyanbrataIISc/TCN-project>.

## References

- [1] Hubel DH, Wiesel TN (1962). Receptive fields, binocular interaction and functional architecture in the cat's visual cortex. *J. Physiol.* **160**, 106–154.
- [2] Hubel DH, Wiesel TN (1968). Receptive fields and functional architecture of monkey striate cortex. *J. Physiol.* **195**, 215–243.
- [3] Ferster D, Miller KD (2000). Neural mechanisms of orientation selectivity in the visual cortex. *Annu. Rev. Neurosci.* **23**, 441–471.
- [4] Maass W (1997). Networks of spiking neurons: the third generation of neural network models. *Neural Netw.* **10**, 1659–1671.
- [5] Gerstner W, Kistler WM (2002). Spiking neuron models: Single neurons, populations, plasticity. *Cambridge University Press*.

- [6] Izhikevich EM (2003). Simple model of spiking neurons. *IEEE Trans. Neural Netw.* **14**, 1569–1572.
- [7] Roy K, Jaiswal A, Panda P (2019). Towards spike-based machine intelligence with neuromorphic computing. *Nature* **575**, 607–617.
- [8] Diehl PU, Neil D, et al. (2015). Fast-classifying, high-accuracy spiking deep networks through weight and threshold balancing. *Proc. IJCNN*, 1–8.
- [9] Rueckauer B, Lungu IA, Hu Y, Pfeiffer M, Liu SC (2017). Conversion of continuous-valued deep networks to efficient event-driven networks for image classification. *Front. Neurosci.* **11**, 682.
- [10] Davies M, Srinivasa N, Lin TH, et al. (2018). Loihi: A neuromorphic manycore processor with on-chip learning. *IEEE Micro* **38**, 82–99.
- [11] Davies M, et al. (2021). Advancing neuromorphic computing with Loihi: A survey of results and outlook. *Proc. IEEE* **109**, 911–934.
- [12] Stimberg M, Brette R, Goodman DF (2019). Brian 2, an intuitive and efficient neural simulator. *eLife* **8**, e47314.
- [13] Tolhurst DJ, Movshon JA, Dean AF (1983). The statistical reliability of signals in single neurons in cat and monkey visual cortex. *Vision Res.* **23**, 775–785.